Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет», ПНИПУ

ЛАБОРАТОРНАЯ РАБОТА

РАБОТА С ОДНОМЕРНЫМИ МАССИВАМИ

Выполнил: студент группы РИС-23-3б

Артем Владимирович Швецов

Проверила: доцент кафедры ИТАС

Ольга Андреевна Полякова

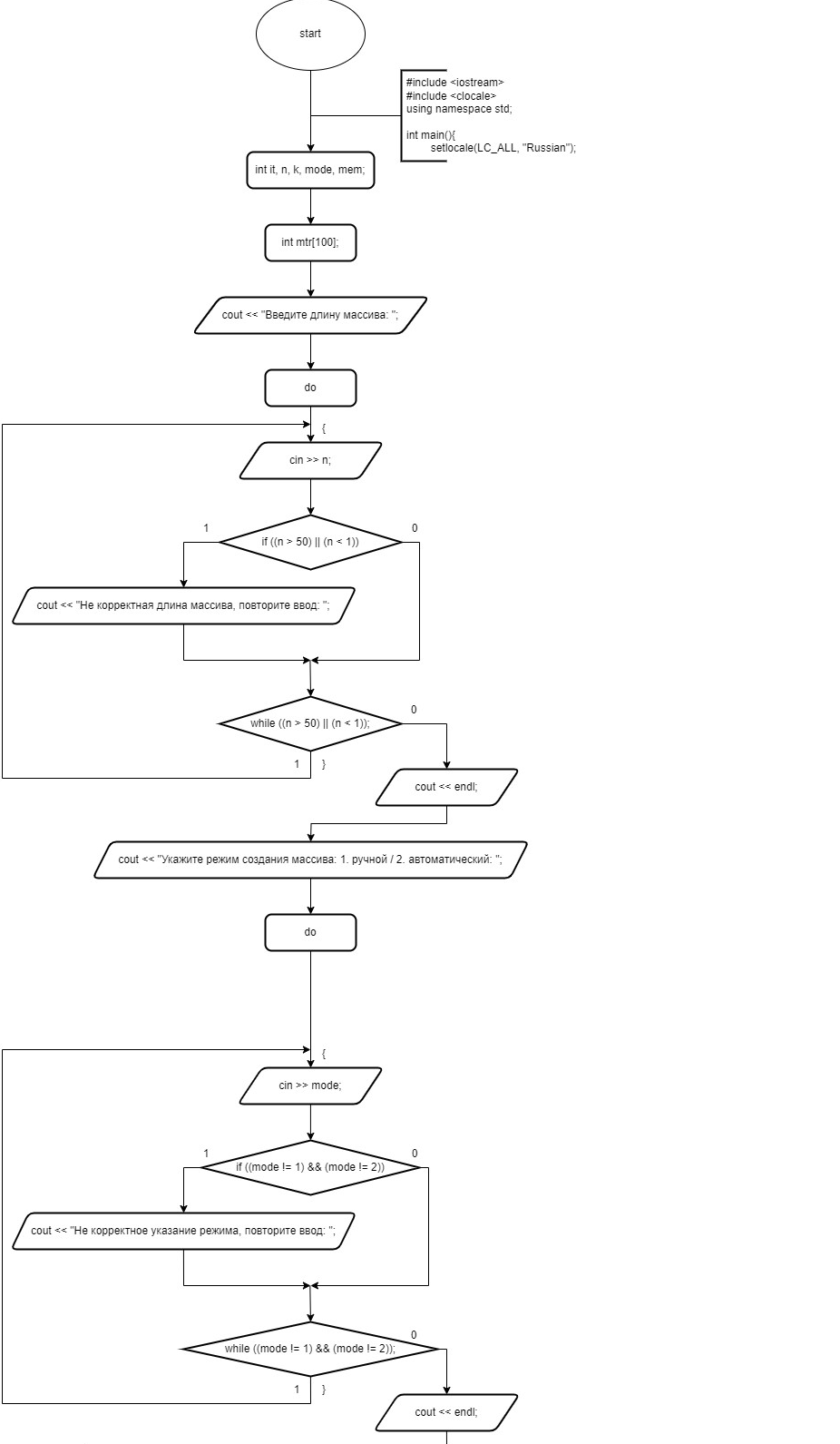
Пермь 2023

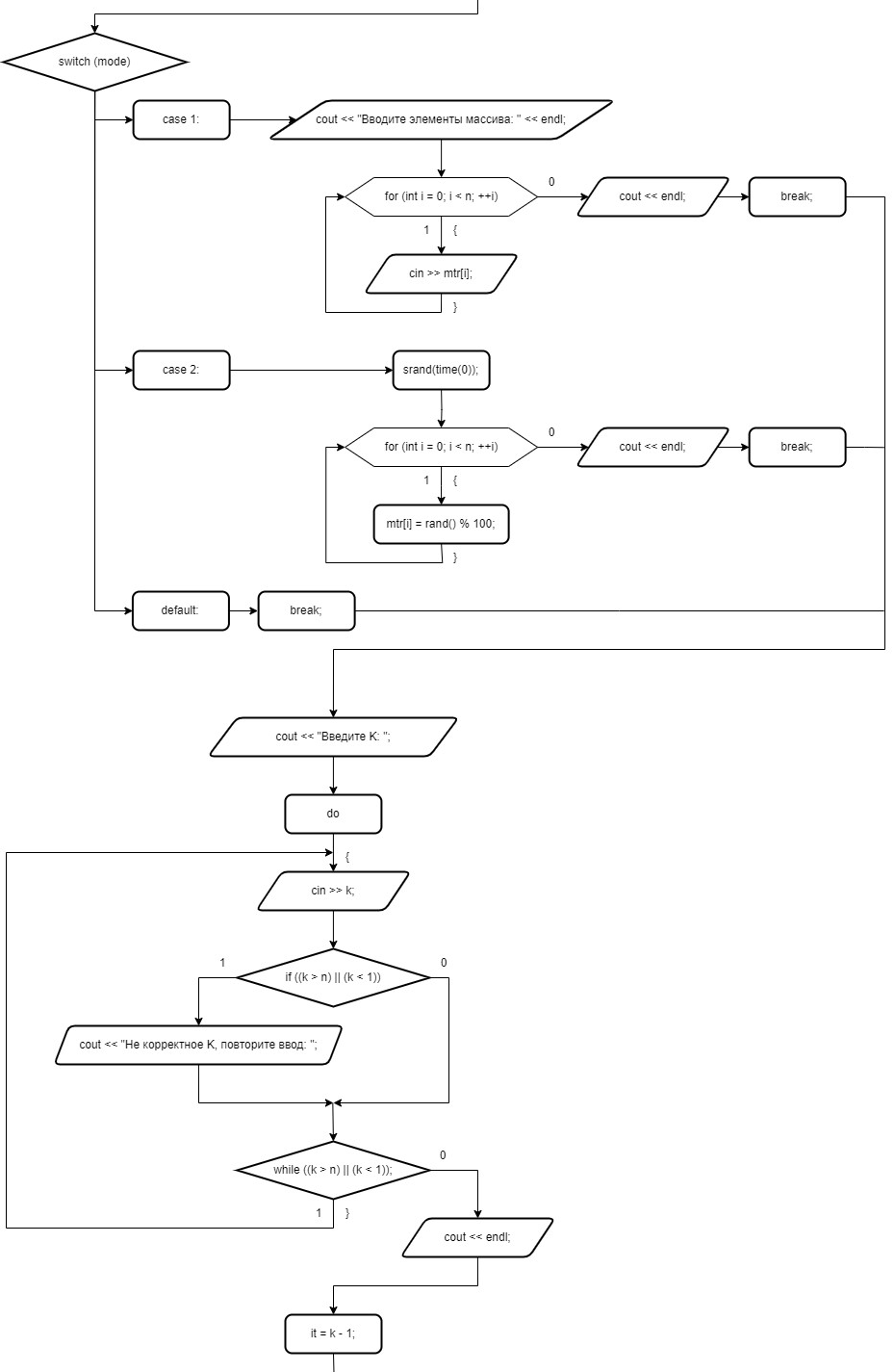
Постановка задачи

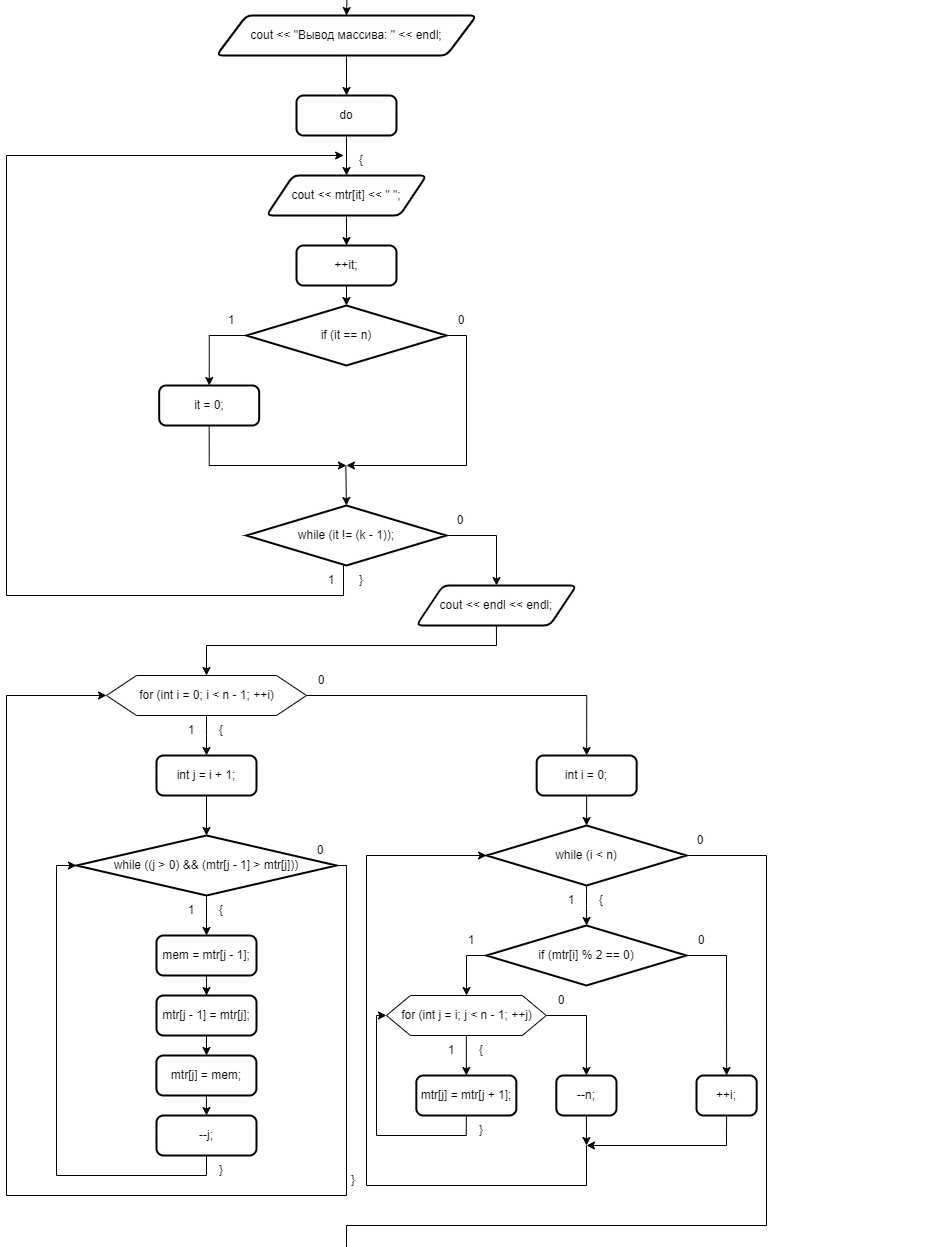
1. Реализовать с использованием массива однонаправленное кольцо (просмотр возможен слева направо, от последнего элемента можно перейти к первому).
2. Распечатать полученный массив, начиная с К-ого элемента и до К-1.
3. Упорядочить элементы по возрастанию.
4. Удалить из кольца четные элементы.
5. Распечатать полученный массив, начиная с К-ого элемента и до К-1.

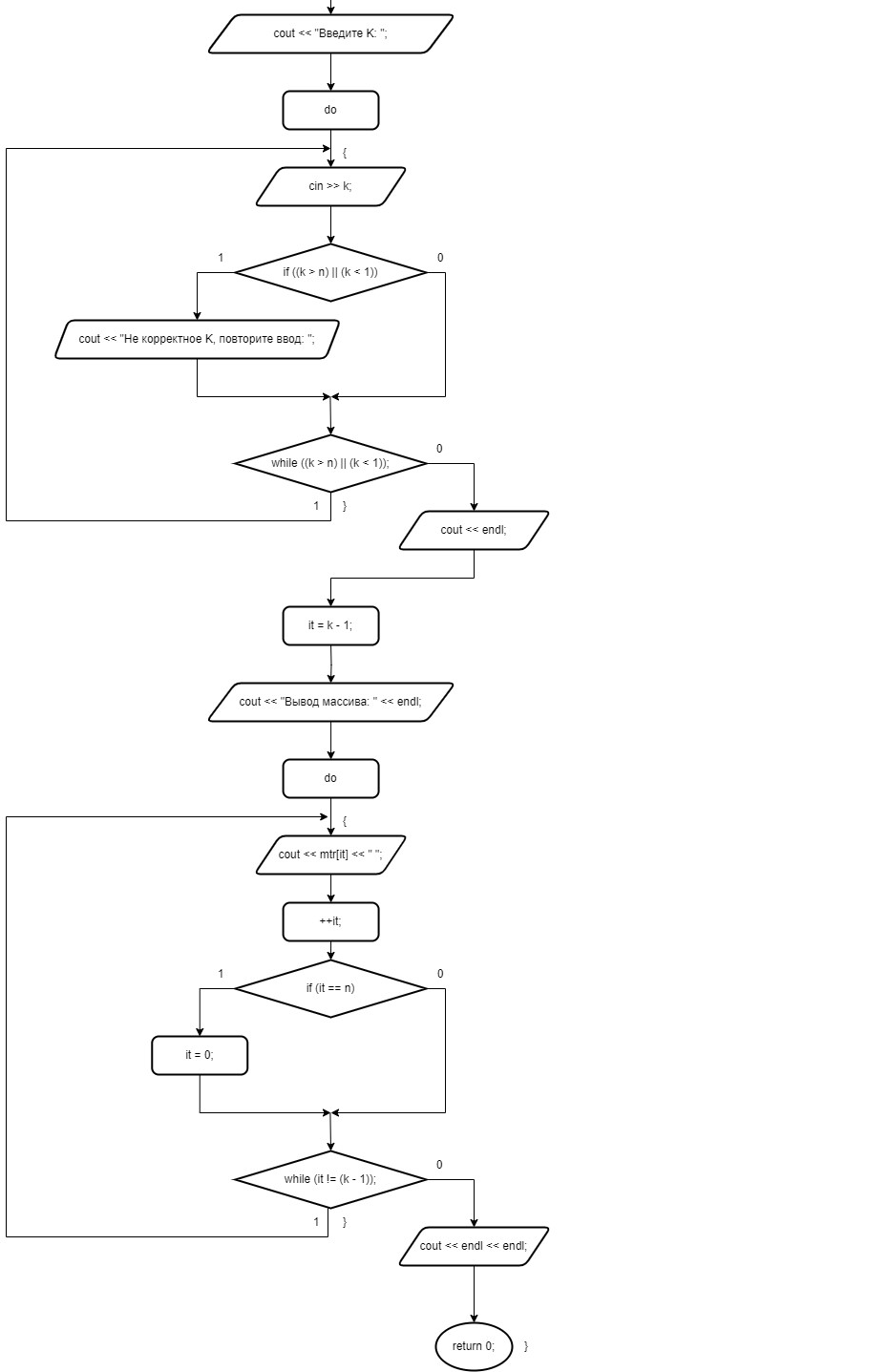
Анализ задачи

1. Перед созданием массива пользователь должен указать длину массива (до 50). Массив формируется в двух режимах: ручной ввод и использование генератора случайных чисел.
2. Для перехода по кольцу используется переменная-итератор. Переход выполняется в итерационном цикле, условие окончания – значение переменной-итератора равно значению К-2. При достижении переменной значения длинны массива ghbcdfbdfybt значения 0.
3. Для отображения массива с K-ого элемента присвоим переменной-итератору значение К-1.
4. Упорядочивание массива будем проводить через метод вставки. Отправная точка – начало массива.
5. Выбор четных элементов проводится проходом справа налево, при нахождении четного числа все последующие элементы сдвигаются на одну позицию влево. Выполняется в итерационном цикле, условие окончания – достижение текущего конца массива.
6. Для повторного отображения массива K задается повторно.

Блок-схема







Код

#include <iostream>

#include <clocale>

using namespace std;

int main() {

setlocale(LC\_ALL, "Russian");

int it, n, k, k\_true, mode, mem;

int mtr[100];

// Указание длины массива в диапазоне от 1 до 50

cout << "Введите длину массива: ";

do {

cin >> n;

if ((n > 50) || (n < 1)) {

cout << "Не корректная длина массива, повторите ввод: ";

}

} while ((n > 50) || (n < 1));

cout << endl;

// Создание массива

cout << "Укажите режим создания массива: 1. ручной / 2. автоматический: ";

do {

cin >> mode;

if ((mode != 1) && (mode != 2)) {

cout << "Не корректное указание режима, повторите ввод: ";

}

} while ((mode != 1) && (mode != 2));

cout << endl;

switch (mode)

{

case 1:

cout << "Вводите элементы массива: " << endl;

for (int i = 0; i < n; ++i) {

cin >> mtr[i];

}

cout << endl;

break;

case 2:

srand(time(0));

for (int i = 0; i < n; ++i) {

mtr[i] = rand() % 100;

}

break;

default:

break;

}

// Отображение массива с К-ого элемента

cout << "Введите K: ";

do {

cin >> k;

if ((k > n) || (k < 1)) {

cout << "Не корректное K, повторите ввод: ";

}

} while ((k > n) || (k < 1));

cout << endl;

it = k - 1; // Получение индекса К-ого элемента

cout << "Вывод массива: " << endl;

do {

cout << mtr[it] << " ";

++it;

if (it == n) {

it = 0;

}

} while (it != (k - 1));

cout << endl << endl;

// Выполненеие сортировки

for (int i = 0; i < n - 1; ++i) {

int j = i + 1;

while ((j > 0) && (mtr[j - 1] > mtr[j])) {

mem = mtr[j - 1];

mtr[j - 1] = mtr[j];

mtr[j] = mem;

--j;

}

}

// Удаление четных элементов

int i = 0;

while (i < n) {

if (mtr[i] % 2 == 0) {

for (int j = i; j < n - 1; ++j) {

mtr[j] = mtr[j + 1];

}

--n;

}

else {

++i;

}

}

// Отображение массива с К-ого элемента

cout << "Введите K: ";

do {

cin >> k;

if ((k > n) || (k < 1)) {

cout << "Не корректное K, повторите ввод: ";

}

} while ((k > n) || (k < 1));

cout << endl;

it = k - 1; // Получение индекса К-ого элемента

cout << "Вывод массива: " << endl;

do {

cout << mtr[it] << " ";

++it;

if (it == n) {

it = 0;

}

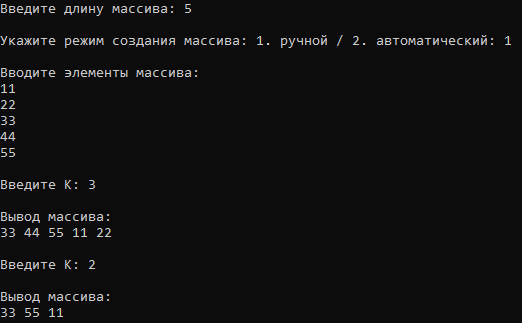
} while (it != (k - 1));

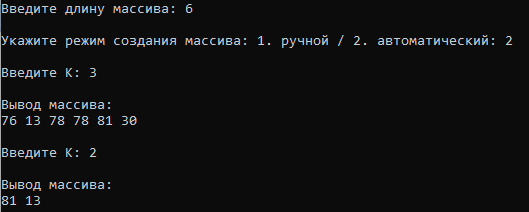
cout << endl << endl;

return 0;

}

Решение





Выводы

В ходе работы, были исследованы специфические методы взаимодействия с массивами: переход по массиву как по кольцу, удаление элементов. Данные методы позволят в будущем более вариативно подходить к решению будущих задач.

Github

